WEST Search History

Hide Items Restore Clear Cancel

DATE: Sunday, March 14, 2004

Hide?	Set Name	Query	Hit Count
	DB=USPT; PLUR=YES; OP=ADJ		
□	L29	(url or uniform resource locator) and l28	40
□	L28	(email or e-mail or e mail or electronic mail) and L27	40
□	L27	(download$ or down-load$ or down load$) and L26	45
□	L26	L25 and rat$4	48
□	L25	detect$ and L24	49
□	L24	L23 and browser and (http or hyper text transfer protocol)	61
□	L23	database and L22	117
□	L22	search$ and L19	122
□	L21	search$ and L19	122
□	L20	search$ and (database ot data base) and L19	0
□	L19	client and server and L18	128
□	L18	select$ and display$ and dictionary and L17	230
□	L17	document same object$ same list$	1380
□	L16	documnet same object$ same list$	0
□	L15	l7 and projector	3
□	L14	internet and L11 and projector	0
□	L13	internet and L11 and projector	0
□	L12	internet and L11	4
□	L11	l7 and L10	5
□	L10	(Genrat$ or updat$) and l5	658
□	L9	http and L8	1
□	L8	L7 and image	5
□	L7	attendee and L6	8
□	L6	L5 and client and server	479
□	L5	(multimedia same presentation) and network and electronic	963
□	L4	image$ and L2	0
□	L3	image and L2	0
□	L2	6167395.pn.	1
□	L1	6167295.pn.	1

END OF SEARCH HISTORY

First Hit Fwd Refs

End of Result Set

□ Generate Collection Print

L3: Entry 1 of 1 File: USPT Feb 4, 2003

DOCUMENT-IDENTIFIER: US 6516337 B1
TITLE: Sending to a central indexing site meta data or signatures from objects on a computer network

Brief Summary Text (10):
Another inherent shortcoming of the method of indexing utilized in the conventional search engine 10 is that only Standard General Markup Language (SGML) information is utilized in generating the central index. In other words, the spider accesses or renders a respective Web page and parses only the SGML information in that Web page in generating the corresponding portion of the central index. As will be understood by those skilled in the art, due to the format of an SGML Web page, certain types of information may not be placed in the SGML document. For example, conceptual information such as the intended audience's demographics and geographic information may not be placed in an assigned tag in the SGML document. One skilled in the art will appreciate that such information would be extremely helpful in generating a more accurate index. For example, a person might want to search in a specific geographical area, or within a certain industry. By way of example, assume a person is searching for a red barn manufacturer in a specific geographic area. Because SGML pages have no standard tags for identifying industry type or geographical area, the spider on the server 14 in the conventional search engine 10 does not have such information to utilize in generating the central index. As a result, the conventional search engine 10 would typically list not only manufacturers but would also list the location of picturesque red barns in New England that are of no interest to the searcher.

Brief Summary Text (13):
The AltaVista.RTM. Discovery program includes an indexer component that periodically indexes the local set of data defined by the user and stores pertinent information in its index database to provide data retrieval capability for the system. The program generates a full indexing at the time of installation, and thereafter incremental indexing is performed to lower the overhead on the desktop. In building the local index, the indexer records relevant information, indexes the relevant data set, and saves each instance of all the words of that data, as well as the location and other relevant information. The indexer handles different data types including Office'97 documents, various types of e-mail messages such as Eudora, Netscape, text and PDF files, and various mail and document formats. The indexer also can retrieve the contents of an html page to extract relevant document information and index the document so that subsequent search queries may be applied on browsed documents.

Brief Summary Text (14):
A program offered by Excite, known as Excite for Web Servers ("EWS"), gives a web server the same advanced search capabilities used by the Excite search engine on the Internet. This program generates a local search index of pages on the web server, allows visitors to the web server to apply search queries, and returns a list of documents ranked by confidence in response to the search queries. Since the program resides on the web server, even complex searches are performed relatively quickly because the local search index is small relative to the index created by

conventional search engines on the Internet.

Brief Summary Text (18):
Another virus of this type is known as the "W97M/Marker.C." This Word 97 macro virus affects documents and templates and grows in size by virtue of tracking infections along the way and appending the victim's name as comments to the virus code. Files are written to the hard drive on infected systems: one file prefixed by C:.backslash.HSF and then followed by random generated eight characters and the .SYS extension, and another file named "c:.backslash.netldx.vxd". Both files serve as ASCII temporary files. The .SYS file contains the virus code and the .VXD file is a script file to be used with FTP.EXE in command line mode. This ftp script file above is then executed in a shell command sending the virus code which now contains information about the infected computer to the virus author's web site called "CodeBreakers."

Detailed Description Text (58):
In a conventional search engine, the search engine normally requests that a web server deliver HTML documents to the search engine, regardless of whether the contents of the page have changed since the last recursive search. This is wasteful not only of CPU resources, but very wasteful of bandwidth which is frequently the most valuable resource associated with a web site. Thus, current search engines and content directories require regular retrieval and parsing of internet-based documents such as web pages. Most search engines use a recursive retrieval technique to retrieve and index the web pages, indexing first the web page retrieved and then all or some of the pages referenced by that web page. At present, these methods are very inefficient because no attempt is made to determine if the information has changed since the last time the information was retrieved, and no map of the information storage is available. For example, a web server does not provide a list of the available URLs for a given web site or series of sites stored on the server. Secondly and most importantly, the web server does not provide a digital signature of the pages available which could be used to determine if the actual page contents have changed since the last retrieval.

Detailed Description Text (62):
Each of these methods relies on duplicating the remote data, which can present difficulties. For example, redundant hardware at the remote and central locations must be purchased and maintained for the storage and transfer of the data over the intranet. Data concurrency problems may also arise should transmission of differential data from the remote locations to the central location be unsuccessful or improperly applied to the central database. Furthermore, if the intranet fails, all operations at remote locations may be forced to cease until communications are reestablished. A further difficulty is the author's loss of authority over his document and the responsibility for retention and data management decisions. In a centralized intranet, unregulated retrieval of objects from the central database to local storage can creates version control problems. Difficulty in handling revisions to an object may also arise in such a centralized system, with simultaneous revision attempts possibly causing data corruption or loss. Finally, in centralized system the size of the central database can grow to the point where management of the data becomes problematic.

Detailed Description Text (155):
Referring to FIG. 23, the document-related packages, com.activeindexing.doc.html, contains classes related to HTML tokenizing and parsing, as shown in more detail in FIG. 36.

Detailed Description Paragraph Table (6):
TABLE 6 Agent Created Articles & Documents Table 13. vii. Type of Articles or Documents 14. Category - three letters representing General, Specific, and Special Interest Categories 15. Related Categories 1, 2, 3, 4, 5, 6, 7, 8, 9 & 10 16. Subject of Articles or Documents 17. Site URL, 18. Unique Record Identifier 19.

viii. Date 20. ix. Author 21. x. Source of Articles or Documents 22. 23. 24. Link URL Link Table

Detailed Description Paragraph Table (38):
TABLE 38 Name Package Description Access com.activeindexing. Contains classes related to the server.access UserAccessService. Agent com.activeindexing. Contains the agent application logical agent control classes. Brochure com.activeindexing. Contains classes related to brochure shared.brochure handling. Catalog com.activeindexing. Contains classes related to the agent shared.catalog CatalogManager. Config con.activeindexing. Contains classes related to util.config Database com.activeindexing. Contains classes related to database server.database access and record handling. HTML com.activeindexing. Contains classes related HTML token- doc.html izing and parsing. Index com.activeindexing. Contains the IndexSegmentService and shared.index related index support classes. package Jini com.activeindexing. Contains classes related to util.jini I/O com.activeindexing. Contains utility classes related to input/ io package output operations. Log com.activeindexing. Contains classes related to the log files. log Message com.activeindexing. Contains classes related to the shared.message MessageQueueService Net com.activeindexing. Contains utility classes related to net networking. Query com.activeindexing. Contains classes related to the server.guery QueryDispatchService. Rating com.activeindexing. Contains classes related to system con- shared.rating figuration file handling. Report com.activeindexing. Contains classes related to report doc.report documents. Schedule com.activeindexing. Contains classes related to the shared,schedule ScheduleManager Servlet com.activeindexing. Contains classes related to Servlets and server.servlet web servers. Signature com.activeindexing. Contains classes related to the file shared.signature signatures and hash calculations. Snmp com.activeindexing. Contains classes related to SNMP util.snmp (Simple Network Management Protocol). Update com.activeindexing. Contains classes related to the server.update UpdateManagerService. Validate com.activeindexing. Contains classes related to data shared.validate validation. XML com.activeindexing. Contains classes designed to help work doc.xml with the DOM (Document Object Model) and SAX interfaces

Detailed Description Paragraph Table (45):
TABLE 45 Class Description BrochureService This is an implementation of a service that provides access to brochures on the server. It is used by the servlets to provide brochure management services and by the update manager to verify content. BrochureDocument A brochure document is an XML representation of a brochure. DocumentBrochure A document brochure applies to html documents. DatabaseBrochure A database brochure applies to databases on the target machine.

Detailed Description Paragraph Table (47):
TABLE 47 Class Description IndexSegmentService An index segment is a pieces of the master index constrained to a range of entries for performance optimization. A range is defined by the IndexSegmentRange class and the index is kept in memory. This class exposes a Jini service for dynamic availability reasons. IndexEntry An index entry contains an identifier, reference to a content page, field reference, hit count and context flags. IndexField A field entry contains only an identifier and text name. It is used for database normalization by the index entries. IndexPage A page reference contains a document identifier, URL to the indexed page, a signature key, mime type, modification date, title, description and index file reference. IndexContext A context defines a position where the index entry was found, either i the title, meta information or in the body of the document. IndexInputStream This stream provides utility functionality to make it easier to read index objects from an input device. IndexOutputStream This stream provides utility functionality to make it easier to write index objects to an output device. IndexSegmentRange This class encapsulates a segment range, which is defined by two string values representing the from and to tokens.

Detailed Description Paragraph Table (54):

TABLE 54 Class Description ReportTemplate Template defining document format for reports, in- cluding field placement information and header, footer specifications. ReportManager High level control class for generating reports. ReportDocument Report document definition.

Detailed Description Paragraph Table (55):
TABLE 55 Class Description XMLManager This class provides access to high-level document control for reading and writing DOM objects. XMLDocumentList This Class provides a mechanism for handling collections of XML documents. XMLUtilities There are numerous operations which are common but not straight forward with the Document Object Model. This class provides a collection of methods to make working with DOM object easier.